Predicting Performance and Creating Better Student Proficiency Models by Improving Skill Codings

Elizabeth Ayers
Thesis Proposal

September 4, 2007

Abstract

Interest in end-of-year accountability exams has increased dramatically since the passing of the NCLB law in 2001. This push has impacted educational research in a wide variety of ways, including a strong desire to be able to model student work in order to make conclusive statements about what students know and how this relates to how they will perform on end-of-year standardized exams. This thesis will look at using item response theory (IRT) to estimate student proficiency. This estimated proficiency will then be used to build prediction models for end-of-year exam scores. Next, methods to improve a skills model will be explored. Models that account for learning over time will then be considered. Finally, I will compare various different approaches to modeling response data.

1 Introduction

Since the passing of the United States Public Law 107-110 (the No Child Left Behind Act of 2001, NCLB) there has been a push within education to raise the standardized test scores of students. As a consequence many students are being given more benchmark exams or assignments during the year in an attempt to uncover their knowledge and understanding. This push has impacted educational research in a wide variety of ways, including a strong desire to be able to model student work in order to make conclusive statements about what students know and how this relates to how they will perform on end-of-year standardized exams. However, due to limited classroom time, teachers must choose between time spent assessing students to answer the above questions and time spent teaching. Educational research can help solve this dilemma by developing reliable tools to model and estimate student knowledge and predict performance.

In this thesis I will explore methods to better model, estimate, and understand student knowledge and make predictions about end-of-year exam performance. I will first discuss the areas in which I have already done work and will give an overview of areas in which I plan to work in the future. I will review current methodology and consider changes, improvements, and extensions.

The methods within this proposal are demonstrated using data from an on-line Mathematics tutor known as the Assistment System (Heffernan et al., 2001; Junker, 2006). During the 2004–2005 school year, over 900 eighth-grade students in Massachusetts used the tutor to prepare for the Massachusetts Comprehensive Assessment System (MCAS) Exam. The MCAS exam is part of the accountability system that Massachusetts uses to evaluate schools and satisfy the requirements of the 2001 NCLB law[1].

In Section 2, I will describe the work I have done in building prediction models. I will begin with a discussion of Item Response Theory (IRT; van der Linden and Hambleton, 1997) and the benefits of using it to estimate student proficiency. I will compare two different models, the Rasch model (Fischer and Molenaar, 1995) and the Linear Logistic Test Model (LLTM; Fischer, 1974). I will then explore the use of an estimated proficiency in predicting end-of-year exam scores. These prediction models will then be compared to other models that use a percent correct as an estimate of student understanding to show that the extra time spent estimating an IRT student proficiency (compared to calculating percent correct) is worthwhile since it leads to better performing prediction models.

When discussing what students know, educational researchers often refer to the Q-matrix (Embretson, 1984), which tags problems with specific skills or knowledge components. In Section 2 I use a specific Q-matrix designed by colleagues working on the Assistment Project. However, there is evidence that this particular Q-matrix is not sufficient. In Section 3, I will examine methods to improve the Q-matrix. I will present a data-driven approach that evaluates problem difficulty estimates and can automatically suggest skills that need to be further explored. This section contains the work that I am currently doing.

The last two sections give a brief description of work that I will do in the next year. In Section 4, I will account for student learning over time. Over the course of the year it is reasonable to assume that a student's proficiency is changing, however in the IRT models that I have used so far there is only a single proficiency estimate. I will explore the use of a multidimensional Rasch model to account for learning over time.

In Section 5, I will compare different modeling approaches including those already mentioned, a multidimensional IRT (MIRT; Embretson, 1991), and the Deterministic Inputs, Noisy "And" Gate (DINA; Junker and Sijtsma, 2001) model. In MIRT a different ability parameter is estimated for each different ability within

[1] See more at http://www.doe.mass.edu/mcas.

a domain. For example, instead of estimating an ability parameter for math, one could estimate separate ability parameters for algebra, geometry, etc. The DINA model makes different assumptions (than the LLTM) about the way in which skills interact in predicting student responses.

2 IRT and Prediction Functions

2.1 Motivation

With the increased interest in standardized testing there has been an increased interest in predicting student performance on end-of-year exams from work done throughout the year (Olson, 2005). When predicting end-of-year exam performance, one of the most commonly used sources of student work is benchmark exams. A common measure of student understanding for many researchers is percent or number correct (e.g., Nuthall and Alton-Lee, 1995; Maccini and Hughes, 2000). Many popular prediction methods use a simple percent correct or number of correct problems on the exams as a factor in prediction models (Bishop, 1998; Haist et al., 2003). However, one drawback of prediction models of this form is that they do not take into account the difficulty of the problems. For example, if two students see different sets of 10 problems and both correctly answer seven, we should be cautious about using percent (or number) correct to compare the students. If one set of problems is harder than the other, then there is an obvious difference of abilities.

As a solution to this problem, one can use Item Response Theory (IRT; e.g. van der Linden and Hambleton, 1997) which relates student and problem characteristics to item responses. By separating the problem difficulty from student ability, we can estimate the student's true underlying ability no matter what set of problems they may see. One of the simplest IRT models is the Rasch model (Fischer and Molenaar, 1995), which models student i's dichotomous response (0 = wrong, 1 = correct) to problem j, X_{ij}, in terms of student proficiency (θ_i) and problem difficulty (β_j) as

$$P_j(\theta_i) = P(X_{ij} = 1|\theta_i, \beta_j) = \frac{e^{\theta_i - \beta_j}}{1 + e^{\theta_i - \beta_j}}. \tag{1}$$

When two students take different benchmark tests, the test characteristic functions (the average of the probabilities in Equation 1, $\overline{P}(\theta) = \frac{1}{J}\sum_{j=1}^{J} P_j(\theta_i)$) will be different, depending on the difficulty of the items in the two tests. Then the MLE $\hat{\theta} = \overline{P}^{-1}(\overline{X})$ will automatically adjust estimated proficiency for the differing difficulty of the items on the two benchmark tests, even if $\overline{X}$ is the same for both students. Thus, the IRT estimate of student proficiency is scaled according to the difficulty of the problems that the student saw.

One could then use this student proficiency estimate (in place of percent correct) to build prediction models of the form

$$Z_i = \lambda_0 + \lambda_1 \cdot \theta_i + \sum_{m=2}^{M} \lambda_m \cdot Y_{im} + \epsilon_i, \tag{2}$$

where Z_i is student i's score on the end-of-year exam, θ_i is student i's estimated IRT proficiency, and Y_{im} are other variables used in the regression such as subject or school level background variables and other measures of performance. This approach is similar to the IRT-based errors-in-variables regression model used by Schofield, Taylor, and Junker (2006) in public policy.

A potentially major source of prediction error in Equation 2 is the measurement error in estimating θ_i. Finding the IRT model that best estimates θ_i is a matter of finding the trade-off between better fit (which tends to reduce statistical bias) and the complexity (which tends to increase statistical uncertainty) of the model.

For example, the many individual problem difficulty parameters in the Rasch model tend to enhance model fit while adding to the model's complexity. However, if we know what skills are involved in the problems we can model problem difficulty in terms of the skills, as in the Linear Logistic Test Model (LLTM; Fischer, 1974). This typically reduces the number of parameters (the complexity) at the expense of decreasing the fit of the model. By improving the fit-complexity trade-off we can make more accurate predictions (e.g. lower mean squared error) of end-of-year (accountability) exam scores from benchmark testing.

2.2 Fitting the IRT Models

We know (Massachusetts Dept of Education, 2004) that MCAS multiple choice questions are scaled for operational use with the 3-Parameter Logistic (3PL) model and short answer questions are scaled using the 2-Parameter Logistic (2PL) model from IRT (van der Linden and Hambleton, 1997). We know that Assistment main questions are built to parallel MCAS exam questions and so it might be reasonable to model Assistment main questions using the same IRT models. However, for simplicity the Rasch model (the 1-Parameter Logistic), Equation 1, was used. There is evidence that student proficiencies and problem difficulties have similar estimates under the 3PL and the Rasch model (Wright, 1995) and so we are not losing much information by starting with the Rasch model.

As briefly mentioned in Section 2.1, the many individual problem difficulty parameters in the Rasch model tend to enhance model fit while increasing the model's complexity. At the expense of decreased model fit, we can reduce the number of parameters in estimating θ_i by using the LLTM (Fischer, 1974) which constrains the Rasch problem difficulty parameters accounting to skills in the the Q-matrix. In the LLTM, it is assumed that skill requirements for each problem combine additively to influence problem difficulty. Thus, to use the LLTM we need an account of what skills problems do and do not depend upon. These dependencies can then be assembled into a Q-matrix (Embretson, 1984; Tatsuoka, 1995; cf. Barnes, 2003 for a recent, more-elaborate application in intelligent tutoring). The Q-matrix, also referred to as a transfer model or skill coding, is a matrix

$$Q = \begin{bmatrix} q_{1,1} & q_{1,2} & \cdots & q_{1,K} \\ \vdots & \ddots & & \vdots \\ q_{J,1} & q_{J,2} & \cdots & q_{J,K} \end{bmatrix},$$

where $q_{jk} = 1$ if problem j contains skill k and 0 if it does not. Thus, the Q-matrix simply indicates which skills each problem depends on. Combining this information, we have the LLTM

$$P_j(\theta_i) = P(X_{ij} = 1|\theta_i, \alpha_k) = \frac{e^{\theta_i - \sum_{k=1}^{K} q_{jk}\alpha_k}}{1 + e^{\theta_i - \sum_{k=1}^{K} q_{jk}\alpha_k}}. \tag{3}$$

The reader may note that we have not included the normalization constant c (Bechger, Verstralen, and Verhelst, 2002) in our representation of the LLTM. This decision will be explained when estimation is discussed. In Equation 3, θ_i is again the proficiency of student i. Here K is the total number of skills in the Q-matrix being used and the q_{jk} are the entries of that Q-matrix . Thus, β_j from Equation 1 is now a linear combination of the skills that appear in problem j. The α_k represents the difficulty of skill k. When there are fewer skills K than test problems J, the LLTM is a restricted form of the Rasch model: for example, if the Q-matrix is the $J \times J$ identity matrix, we obtain the unrestricted Rasch model again. The LLTM has been successful in other works such as van de Vijver (1988) and De Boeck and Wilson (2004). In both of these cases, the correlation between the Rasch model and LLTM problem difficulties was greater than 0.90.

The dichotomous responses X_{ij} are modeled as Bernoulli trials,

$$X_{ij} \sim Bern(P_j(\theta_i)) \qquad i = 1, \ldots, N;\ j = 1, \ldots, J,$$

Table 1: Rasch vs LLTM fits for Assistment Main Questions

Model	$-2 \cdot l_M$ = Deviance	Parameters	BIC
LLTM	56090	79	~ 56605
Rasch	47640	356	~ 49963
		Difference in BIC	~ 6600

where $P_j(\theta_i)$ is given above by Equation 1 or Equation 3. Under the usual IRT assumption of independence between students and between responses, given the model parameters, the complete data likelihood is

$$P(\underline{X} = \underline{x}) = \prod_{i=1}^{N} \prod_{j:i \text{ saw } j} P_j(\theta_i)^{x_{ij}} [1 - P_j(\theta_i)]^{1-x_{ij}}. \tag{4}$$

Since not all students saw the same set of problems, the second product only includes the set of problems that student i saw. We estimated the student proficiency (θ_i) and problem difficulty (β_j) parameters in the Rasch model and the student proficiency (θ_i) and skill difficulty (α_k) parameters in the LLTM, using Markov Chain Monte Carlo methods with the program WinBUGS[2] (Bayesian inference Using Gibbs Sampling; Spiegelhalter et al., 2003), with the priors $\theta_i \sim N(\mu_\theta, \sigma^2_\theta)$ and $\beta_j \sim N(\mu_\beta, \sigma^2_\beta)$. We placed a weak Normal hyperprior on μ_β and an Inverse-Gamma hyperprior on σ^2_β. In item response models, the location and scale of the latent variable, and hence of problem difficulty parameters, are not fully identified, which can undermine comparisons between fits on different data sets. We decided to fix the (prior) mean and variance of the student proficiency (θ) to be 0.69 and 0.758. These values were found by preliminary analysis using hyperpriors on these parameters. In the LLTM, which constrains the Rasch model as shown by Equation 3, we used the same Normal(0.69,0.758) prior on the student proficiency (θ) and the prior on α was Normal($\mu_\alpha, \sigma^2_\alpha$). Again, a weak Normal hyperprior was placed on μ_α and an Inverse-Gamma on σ^2_α. These priors differ slightly from other estimation methods that use a $N(0, 1)$ prior for θ_i. In our case, one can think of the prior mean on θ taking the place of the normalization constant c.

Recent work (Gelman, 2006) has brought into question the use of an Inverse-Gamma(ϵ, ϵ) hyperprior. In particular, when $\epsilon \to 0$ this prior leads to an improper posterior density. In addition, when low σ values are plausible the prior becomes informative as inferences become sensitive to the value of ϵ. The Inverse-Gamma priors used above $\epsilon = 1$, so the first issue may not be problematic. In any event, additional sensitivity analyses with Gelman's results in mind will be done to complete this portion of the dissertation work.

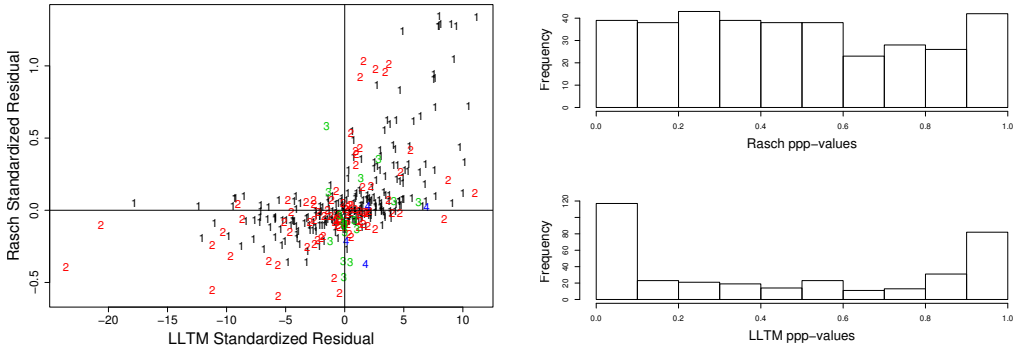
2.3 Direct Model Comparison

We compared the Rasch model and LLTM using Bayesian Information Criterion (BIC; Raftery, 1995) scores,

$$-2 \cdot l_M + k \cdot \log(n). \tag{5}$$

Here l_M is the log-likelihood of the model, k is the number of free parameters to be estimated, and n is the sample size (here, the number of students). In this version of BIC scores, lower values indicate better fitting models and a difference as small as 2 denotes a mentionable difference between models and differences larger than 10 denote a very strong significant difference between the models. WinBUGS tracks the deviance, which is defined as $-2 \cdot l_M$ (Spiegelhalter, Thomas, and Best, 2003), of the model during estimation. Table 1

[2] WinBUGS and R code available from the authors on request.



(a) Assistment residual

(b) Posterior Predictive P-value

Figure 1: Assistment Main Question Residuals and ppp-values. Figure (a) is Number Coded by the Number of Skills in the Problem.

shows the BIC scores for both the Rasch model and LLTM. One can see that the difference in BIC scores is $\sim$ 6600 and thus the Rasch model is overwhelmingly favored. Although the Rasch model is more complex (in the number of parameters) than the LLTM, the dramatically better fit of the Rasch model makes up for the complexity and the Rasch model is strongly favored.

To explore the misfit of the LLTM, we looked at the per problem standardized residuals

$$r_j = \frac{n_j - E(n_j)}{\sqrt{\widehat{Var}\,(n_j)}}. \tag{6}$$

Here, $n_j = \sum_{i:i\ saw\ j} X_{ij}$ is the number of correct answers to problem j, $E[n_j]$ is its expected value estimated from fitting the model in Equation 1 or 3, and $\widehat{Var}\,(n_j)$ is its variance estimated from the same model. We also calculated the per problem outfit statistics (van der Linden and Hambleton, 1997, page 113),

$$T_j(x \mid \phi) = \sum_{i=1}^{N_j} \frac{(X_{ij} - E_{ij})^2}{N_j W_{ij}},$$

where N_j is the number of students that saw problem j, X_{ij} is student i's response on problem j, E_{ij} is the expected value of X_{ij} conditional on the parameter vector ϕ, and W_{ij} is the variance of X_{ij} also conditional on ϕ. To check the per problem fit of each model, the posterior predictive p-value (ppp-value; Gelman et al., 2004), the expected value of the classical p-value over the posterior distribution of the parameter vector given the model and the observed data, was estimated using

$$p_i \approx \frac{\#\{s : T_i(x \mid \phi_x) < T_i(x^* \mid \phi_x);\ s = 1, 2, ...M\}}{M},$$

which compares the observed values of the test statistic to values of the test statistic for data simulated from the model. For this calculation, the simulated data (x^*) was obtained by using the Markov Chain given by WinBUGS. Similar to classical p-values, there is reason to question the fit of the model to problem i if p_i is small. A weakness of the ppp-value is that it uses the data twice, once to calculate the observed test statistics

and again to simulate data to calculate the ppp-value. One consequence of this is that ppp-values are not uniformly distributed and tend to be conservative (Gelman et al, 1996, page 790). However, we can still expect the ppp-values to aggregate around zero if there is serious misfit for some of the problems.

Figure 1 (a) shows the Rasch versus LLTM residuals, as described in Equation 6. The residuals are number coded by the number of skills in the problem. For these questions, the Rasch model residuals (vertical axis) range from −.6 to 1.4, indicating good fit, and the LLTM residuals (horizontal axis) range from −23 to 11.2, indicating bad fit. Figure 1 (b) shows the histograms for the Rasch model and LLTM ppp-values. We see that the Rasch model ppp-values are roughly uniform, which we would expect if the model fit is acceptable. For the LLTM, the grouping of ppp-values around 1 shows the weakness of ppp-values to be bias toward accepting the model. However, there are also many ppp-values concentrated at 0 giving the stronger impression of misfit of the model.

2.4 Reliability and Predictive Accuracy

To compare prediction models we computed the 10-fold cross-validation mean absolute prediction error or the mean absolute deviation,

$$MAD = \text{mean } |\mathrm{Z_i} - \text{predicted } \mathrm{Z_i}| = \frac{1}{\mathrm{N}} \sum_{\mathrm{i=1}}^{\mathrm{N}} |\mathrm{Z_i} - \text{predicted } \mathrm{Z_i}|. \tag{7}$$

MAD is used because it is considered to be more interpretable by the Assistment developers. We also report the cross-validation mean squared error (MSE).

However, before exploring the predictive accuracy of our models using the MAD measure defined in Equation 7, it is important to ask how well Assistment performance could predict MCAS scores under ideal circumstances. Let us begin by assuming the MCAS exam and the Assistment System are two parallel tests of the same underlying construct. Following classical test theory (Lord and Novick, 1968) we have

$$\begin{aligned} X_{i1} &= T_i + \epsilon_{i1} \\ X_{i2} &= T_i + \epsilon_{i2} \end{aligned}$$

where the true score of student i is T_i, X_{it} is student i's observed score on test t, and ϵ_{it} is the error on test t. We have followed the usual assumptions that the expected value of the error terms are zero, the error terms are uncorrelated, and that the error terms and the true score are uncorrelated. The expected mean square error (MSE) between the tests is then

$$E[(X_{i1} - X_{i2})^2] \quad = \quad E[(\epsilon_{i1} - \epsilon_{i2})^2] = \sigma^2_{\epsilon_1} + \sigma^2_{\epsilon_2}.$$

Since the reliability of test t ($t = 1$ or 2) is defined as

$$r_t = \frac{\sigma^2_T}{\sigma^2_{X_t}} = \frac{\sigma^2_T}{\sigma^2_T + \sigma^2_{\epsilon_t}}, \tag{8}$$

some algebra then shows that the root mean square error (RMSE) is

$$RMSE = \sqrt{E[(X_{i1} - X_{i2})^2]} = \sigma_T \sqrt{\left(\frac{r_1 + r_2}{r_1 \cdot r_2} - 2 \right)}.$$

This can be converted into lower and upper bounds on the MAD score. Using the Cauchy-Schwarz inequality for Euclidean spaces (Protter and Morrey, 1991, page 130) with $x_i = |\text{MCAS}_\text{i} - \text{predicted MCAS}_\text{i}|$ and $y_i = 1$,

$$\sum_{i=1}^{N} |\text{MCAS}_\text{i} - \text{predicted MCAS}_\text{i}| \leq \sqrt{N} \cdot \sqrt{\sum_{i=1}^{N} (\text{MCAS}_\text{i} - \text{predicted MCAS}_\text{i})^2}.$$

We can then scale both sides by $\frac{1}{N}$ to achieve

$$MAD \leq RMSE.$$

To bound the MAD from below let $x_i = \text{MCAS}_\text{i} - \text{predicted MCAS}_\text{i}$ and $|x_{max}|$ denote the absolute maximum deviation between the true and predicted MCAS scores. Then,

$$RMSE^2 = \frac{1}{n} \sum_{i=1}^{N} \text{x}_\text{i}^2 \leq \frac{1}{\text{n}} \sum_{\text{i}=1}^{\text{N}} |\text{x}_\text{i}| \cdot |\text{x}_\text{max}| = |\text{x}_\text{max}| \frac{1}{\text{n}} \sum_{\text{i}=1}^{\text{N}} |\text{x}_\text{i}| = |\text{x}_\text{max}| \text{MAD},$$

so we have that

$$\frac{1}{|x_{max}|} \cdot RMSE^2 \leq MAD.$$

Thus, our lower and upper bounds for the MAD score are

$$\frac{1}{|x_{max}|} \cdot RMSE^2 \leq MAD \leq RMSE. \tag{9}$$

From Equation 8, we have that $\sigma_T^2 = r_t \cdot \sigma_X^2$. In the most recent technical report published (Massachusetts Dept of Education, 2006) the MCAS has listed $r_{t=1} = 0.9190$ and $\sigma_X^2 = 142.39$, so that in predicting MCAS exam scores from Assistment scores we have

$$RMSE = \sqrt{130.86 \cdot \left(\frac{0.9190 + r_2}{0.9190 \cdot r_2} - 2 \right)}, \tag{10}$$

where r_2 is the reliability of the Assistment score.

However, since each student completes a unique set of Assistment questions, we could not calculate a single r_2 directly. Instead, we calculated reliability separately for each student. For this purpose we considered a reduced dataset of 616 students who had 10 or more problems completed for which all pairs of correlations were available. To estimate the per-student reliability, we used Cronbach's alpha coefficient (Cronbach, 1951),

$$\alpha_i = \frac{n_i \overline{r}_i}{1 + (n_i - 1)\overline{r}_i}. \tag{11}$$

In Equation 11, n_i is the number of problems seen by student i and $\overline{r}_i$ is the average inter-item correlation for problems seen by student i. Once per-student reliabilities were calculated, the per-student estimated RMSE values were computed using Equation 10. Figure 2 shows the estimated reliabilities for the students who met the criteria explained above. It is interesting to note that the estimated RMSE is never lower than 4.44.

In order to have a single approximate set of approximate bounds for the MAD score in Equation 9, we found the median Assistment reliability, 0.8080, and the corresponding RMSE of 6.529 from Equation 10. The largest deviation, $|x_{max}|$, between the true and predicted MCAS scores among the models in Table 2 below was 40.5. Substituting these values for RMSE and $|x_{max}|$ into Equation 9 we find the approximate bounds,

$$1.053 \leq MAD \leq 6.529.$$

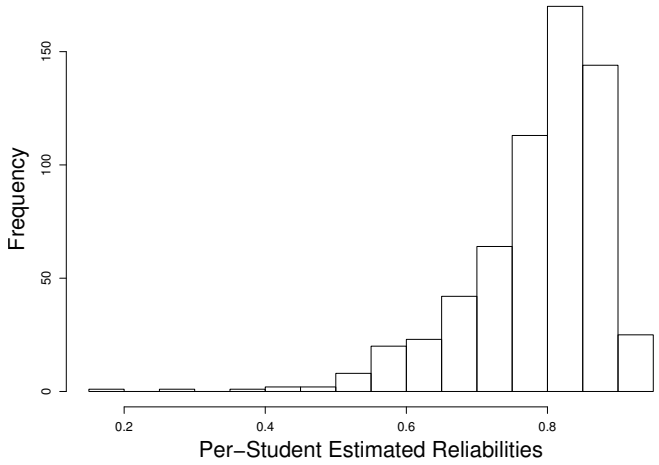


Figure 2: Histogram of per-student Assistment reliabilities as given by Equation 11

2.5 MCAS Exam Score Prediction

Student proficiency estimated from a successful IRT model is combined with other Assistment performance metrics to produce an effective prediction function, following the work of Anozie and Junker (2006), using an errors-in-variables regression approach similar to that of Schofield et al. (2005). The linear model is

$$MCAS_i = \lambda_0 + \lambda_1 \cdot \theta_i + \sum_{m=2}^{M} \lambda_m \cdot Y_{im} + \epsilon_i,$$

where θ_i is the proficiency of student i as estimated by the IRT model and Y_{im} is performance of student i on manifest measure m. WinBUGS was again used to find Bayesian estimates of the linear regression coefficients. When estimating each of the following models, the IRT item parameters were fixed at their estimates from Section 2.2, but student proficiency was re-estimated. It is logical to not re-estimate item parameters since this is how MCAS prediction would occur in practice: problems are fixed but student proficiencies are changing throughout the year and year-to-year.

Table 2 shows results from several prediction models. In view of the results of Section 2.3, the IRT based prediction models below are based on the Rasch model estimate of student proficiency. Column 2 lists which variables are in the model (for a full list and description of the variables see Table 3). Column 3 simply states the number of variables in the model. Columns 4 and 5 give the CV MAD score and the CV RMSE respectively. Column 6 offers some important notes about the models. Historically, and in particular within the Assistment Project, percent correct on questions has been used as a proxy for student ability. To see if information is gained by using the Rasch estimate of student proficiency, we compared the two models with only these variables. Model 1 is the simple linear regression using only percent correct and has a MAD score of 7.18. Model 2 uses only the Rasch student proficiency and gives a MAD score of 5.90. By simply using IRT to account for problem difficulty in estimating student proficiency, we can drop the MAD score a full point. Accounting for problem difficulty gives a more efficient estimate of how well a student is doing and leads to better predictions. Model 3, from Anozie and Junker (2006), uses as predictors monthly summaries from October to April for percent correct on questions and four other manifest measures of student performance. Model 4 uses the year-end aggregates of the same variables and substitutes Rasch

Table 2: Prediction Models

Model	Variables	# of Vars	CV MAD	CV RMSE	Notes
Model 1	Percent Correct on main questions	1	7.18	8.65	
Model 2	Rasch student proficiency	1	5.90	7.18	
Model 3 (Anozie & Junker, 2006)	Percent Correct on main questions and 4 other manifest performance metrics	35	5.46	7.00	uses multiple monthly summaries
Model 4	Rasch student proficiency and same 4 manifest performance measures as Model 3	5	5.39	6.56	uses only year-end aggregates
Model 5	Rasch student proficiency and 5 manifest performance measures (one overlap with models 3 & 4)	6	5.24	6.46	optimized for student proficiency

student proficiency for percent correct on questions. We see that Model 4 gives a slightly lower MAD score. Thus by using Rasch student proficiency (in place of percent correct) we can use fewer, more-aggregated measures of student performance on Assistments.

Model 5 was optimized (for MAD score) for Rasch student proficiency and year end aggregates of student performance measures using backwards variable selection implemented in WinBUGS and R[3] (R Development Core Team, 2004). To start we used the same 12 variables as Anozie and Junker (2006), excluding percent correct on main questions and adding Rasch student proficiency. We ran the full model and all models excluding one variable, with the caveat that student proficiency was always kept in the model. For each model, MCAS exam scores were predicted and MAD scores calculated. The model with the lowest MAD score was then used as the new "full" model. This process was repeated until removing variables from the "full" model no longer reduced the MAD score. The final model, which contained student proficiency and five manifest measures of student performance, gives a MAD score of 5.24, a slight improvement from Model 4. Overall, the ability to use fewer variables makes the effort expended in estimating the IRT models worth it.

The regression equation for Model 2 is

$$MCAS_i = 18.289 + 10.425 \cdot (\text{Rasch student proficiency}). \tag{12}$$

From this we see that there is a baseline MCAS exam score prediction of 18 points and for each additional unit of estimated Rasch student proficiency we add 10.425 to to the exam score prediction. As a student's proficiency increases, so does their exam score prediction. The regression equation for Model 5 is

$$\begin{aligned} MCAS_i &= 8.514 + 10.336 \cdot (\text{Rasch student proficiency}) + 8.928 \cdot (\text{NumPmAllScaf}) \\ &+ 0.004 \cdot (\text{SecCorScaff}) + 0.032 \cdot (\text{MedSecIncMain}) - 0.001 \cdot (\text{SecIncMain}) \\ &- 2.696 \cdot (\text{PctSecIncMain}). \end{aligned} \tag{13}$$

[3]WinBUGS and R code available from the authors on request.

Table 3: Definitions of Variables used in Prediction Models

Variable Name	Model	Definition
Student Proficiency	2, 4, 5	IRT estimate of student Proficiency
PctCorMain	1, 3	Percent of correctly answered main questions
PctCorScaf	3, 4	Percent of correctly answered scaffolds
SecIncScaf	3, 4	Number of seconds spent answering all incorrect scaffolds
NumPmAllScaf	3, 4, 5	Number of scaffolds completed per minute
NumHintsIncMainPerMain	3, 4	$\frac{\text{number hints + number incorrect main questions}}{\text{Number of main questions attempted}}$
SecCorScaff	5	Number of seconds spent answering all correct scaffolds
SecIncMain	5	Number of seconds spent on incorrect main questions
MedSecIncMain	5	Median number of seconds per incorrect main question
PctSecIncMain	5	Percent of time on main q's spend on incorrect main q's

In Equation 13 the increase in MCAS score for each unit of increase in Rasch proficiency is about the same as in Equation 12. However, the baseline of 18.289 has been decomposed into a new baseline of about 8.5 points, incremented or decremented according to various measures of response efficiency. The largest increment, 8.928, comes from the rate at which scaffolding questions are completed and the largest decrement, 2.696, comes from the total amount of time spent on answering main questions incorrectly.

Now that we have compared models to one another, we need to compare the models to the bounds calculated in Section 2.4. Recall from Section 2.4 that we have a bound of

$$1.053 \leq MAD \leq 6.529.$$

From Table 2, one can see that Model 5 has a MAD score of 5.24, which is well below the upper bound.

Moreover, the RMSE reported for Model 5, 6.46, is similar to our estimated optimal RMSE of 6.53. It should also be noted, that with a perfect Assistment reliability in Equation 10, the estimated RMSE would be 5.576 and the bound would be

$$0.768 \leq MAD \leq 5.576.$$

Again, the Model 5 MAD score is below this upper bound. Using a split-half reliability on the MCAS exam itself, Feng et al. (2006) estimated the best-possible MAD for predicting MCAS from Assistments data to be about 5.94. Since we are already achieving MAD scores less than this and the two previously mentioned upper bounds, we do not expect to do much better without an increase in the reliability of the MCAS exam.

2.6 Conclusions

In this section I have developed a framework to create prediction functions for end-of-year exam scores using an IRT estimate of student ability based on work done throughout the school year. Although this framework was illustrated using data from an online mathematics tutor, other benchmark work, such as homework or paper and pencil exams, could be used to predict end-of-year exam scores as well.

In addition to developing this general framework, our research generated two additional findings. First, prediction using IRT scores is more effective than prediction using percent correct scores. For example,

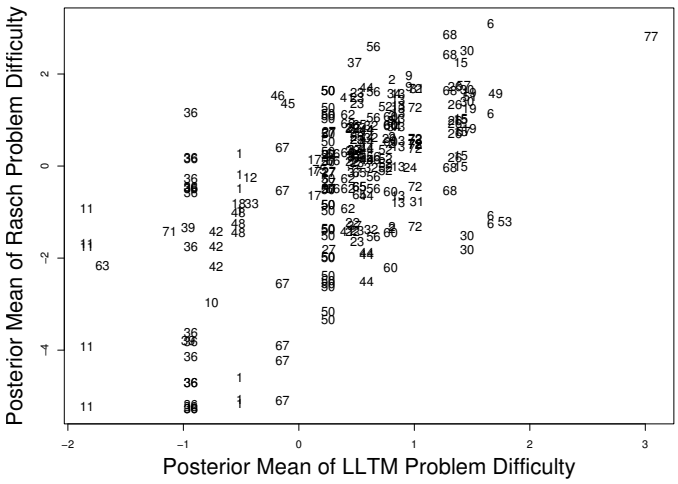


Figure 3: Problem Difficulties for Single Skill Main Questions

our Rasch model based predictions always produced lower MAD and RMSE prediction errors than the corresponding predictions based on percent correct. Moreover, the IRT-based predictions were essentially as good as one could do with parallel tests, even though our Assistment System was not constructed to be parallel (in the classical test theory sense) to the MCAS exam. Second, in our application, the Rasch model outperformed the LLTM. This is contrary to some previous experience with the LLTM (van de Vijver, 1988; De Boeck and Wilson, 2004) and this deserves further exploration.

An obvious place to look is possible improvements in the transfer model (Q-matrix) and Section 3 explores this option. Of course, another possible difficulty may be that the LLTM is inappropriate for this data. Alternative models involving Bayes net style cognitive diagnosis models (Anozie and Junker, 2007; Pardos et al., 2006) and multidimensional IRT models (based on the 5 MCAS strands: Number Sense and Operations; Patterns, Relations, and Algebra; Geometry; Measurement; and Data Analysis, Statistics, and Probability) will be explored in Section 5.

3 Improving the Q-matrix

Until now, the majority of the skill analysis that has been done within the Assistment project has focused on one Q-matrix known as the WPI-April 2005. This skills model has a total of 107 skills (77 of which appear in the problems analyzed in Section 2.3) and was developed by a group of educational researchers at WPI. The skills included in this Q-matrix range from basic skills such as addition or multiplication to more high-level skills such as Pythagorean theorem and stem-and-leaf plot. While many of the researchers have teaching experience, up to now there has been little work done to verify the skills in this model.

In the previous section I proposed two different reasons for the misfit of the LLTM. The first is that the additive assumption of the LLTM is not appropriate and the second is that this Q-matrix does not contain the proper set of skills. The first issue will be explored further in Section 5 where I will compare several different models. In the remainder of this section I will assume that the LLTM is the "right" model and discuss methods to suggest improvements in the Q-matrix. I will first discuss methods that I am using and developing and then compare them to other methods such as Tatsuoka's (1983) rule space method.

Figure 3 shows the Assistment question problem difficulties for single skill problems. The plotted point is a number (from 1-77) which simply identifies which skill is in the problem. In this plot we see several vertical lines of dots which indicate problems with the same skill that have the same difficulty estimate from the LLTM, but different Rasch model estimates. This is an effect of the LLTM since we have forced problems with the same skill to have the same difficulty. There are several skills with noticeable differences between the Rasch model and LLTM estimates. The first is skill 50 (probability), which under the LLTM has a difficulty of 0.25. However,the Rasch model estimates of problems with skill 50 range from -3 to 2. The next skill, number 67 (subtraction), is given a difficulty estimate of -0.14 by the LLTM. In this case, the Rasch estimates of problem difficulties range from -5 to 0. Of particular interest are skills 36 (multiplication) and 1 (addition). The LLTM estimates the difficulty of skill 36 as -0.93. In the Rasch model, problems with this skill are placed into two separate groups, one between -5 and -3.5 and another between -2 and 1. The LLTM estimate of skill 1 is -0.5 and the Rasch again gives two different groups.

In previous work (Ayers and Junker, 2007) we explored using a random effects component (Janssen and De Boeck, 2006) to account for some of this variation. We tried several different priors for the random effect, but in each case the random effect was so large that skills were no longer playing a significant role in modeling problem difficulty. The lack of many skills with difficulty estimates significantly different from zero is an indication of a misalignment between the skills in our model and the difficulty of the problems.

In the discussion of skill 36 above, it was noted that there were two different groups of estimates for skill difficulty. I believe that this suggests a need to split the skill into two different skills. Skills 1 and 67 suggest something similar. Although skill 50 does not have a split, there is a wide range of skill difficulty estimates for the Rasch model and this also suggests a problem.

3.1 Testing

In a series of papers Bechger, Verstralen, and Verhelst (2002, 2004) and Fischer (2004) discussed methods of testing Q-matrix entries. Here I will follow Fischer's work in which he uses a likelihood ratio test to compare the fit of two different Q-matrices. Let q_{jk} be the hypothetical value of a single element of our Q-matrix and σ_{jk} its true value. Our null hypothesis is

$$H_o : \sigma_{jk} = q_{jk}. \tag{14}$$

Under the null, the entire Q-matrix is fixed and the contribution of skill k on item j is $q_{jk}\alpha_k$. Under the alternative hypothesis (of non-equality), the contribution of skill k may vary independently. This contribution can be expressed through the Q-matrix by adding a $K+1$ column e_j (the j^{th} unit vector) so that we have $Q_{test} = (Q, e_j)$. This expression is equivalent to changing the contribution of skill k to item j from $q_{jk}\alpha_k$ to $q_{jk}\alpha_k + \alpha_{K+1}$. This can be rewritten as

$$q_{jk}\alpha_k + \alpha_{K+1} = \sigma_{jk}\alpha_k.$$

In this model, α_{K+1} and σ_{jk} are free parameters to be estimated. The two LLTMs with matrices Q and $Q_{test} = (Q, e_j)$ can be compared using a conditional likelihood ratio (CLR) test with $df = 1$. If the test shows poor fit of Q when compared to Q_{test}, one can estimate σ_{jk} using

$$\hat{\sigma}_{jk} = q_{jk} + \frac{\hat{\alpha}_{K+1}}{\hat{\alpha}_k}, \tag{15}$$

where q_{jk} is from either Q or Q_{test} (as it will be the same in both) and $\hat{\alpha}_{K+1}$ and $\hat{\alpha}_k$ are the skill difficulty estimates from the model using Q_{test}.

Table 4: Fits for Simulated Data

Model	$-2 \cdot l_M$ = Deviance	Parameters	BIC
Q_{mod}	23559.28	6	$\sim 23,600$
Q_{test}	23522.09	7	$\sim 23,570$
Difference in Dev	~ 37	Difference in BIC	~ 30

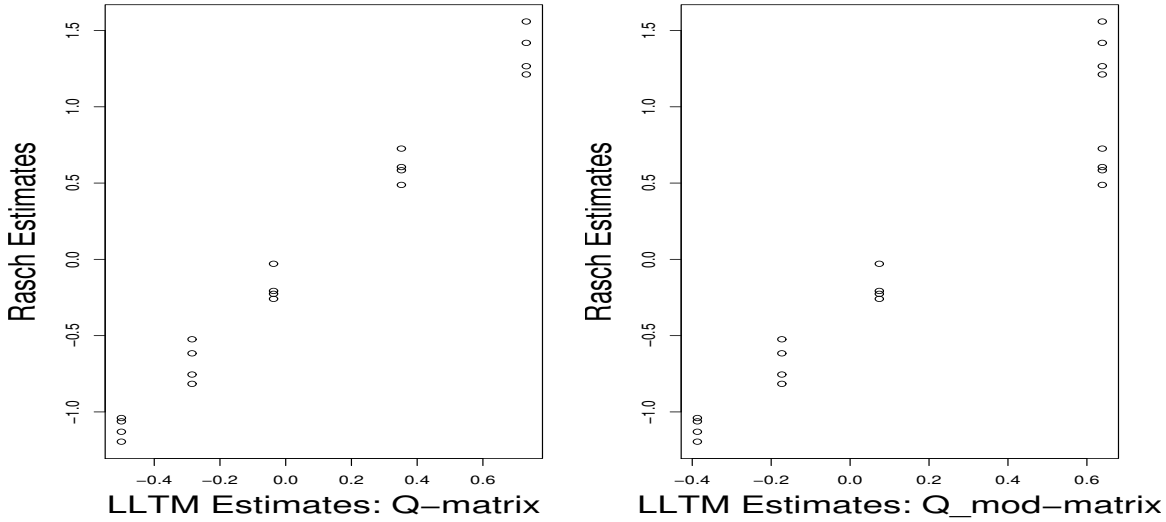


Figure 4: Simulation Study Estimates

It should be noted that in the Bechger, Verstralen, and Verhelst (2002, 2004) and Fischer (2004) papers, the entries of the Q-matrix were not restricted to be 0/1. In their framework, one can consider the $(j,k)^{th}$ entry of the Q-matrix the number of times that skill k is used is answering question j. In the work that I have done, I constrained the entries of the Q-matrix to be 0/1 since this follows the format of the Q-matrices used within the Assistment project. Under this, $q_{j,k}$ only tells us whether or not problem j requires skill k.

Currently I am doing work to investigate modifying the theory explained above to test a Q-matrix with 0/1 entries. In particular, I am exploring the idea that, given a poor fit of Q and an estimation of $\hat{\sigma}_{jk}$ using Equation 15, a value larger than 1 tells us something about Q-matrix. Ideally it would be nice if estimated values larger than 1 indicated skills that needed to be split. To test this idea, I am doing a simulation study.

In the simulation study, I started with $N = 1000$ students, $J = 20$ problems, and $K = 5$ skills. In this first simulation the five skill difficulties ranged from -1 to 1.5. I used a simple 20×5 Q-matrix with each problem requiring only one of the five skills with each skill appearing in exactly four problems.[4] Once I had set this "true" Q-matrix, I created a 20×4 Q_{mod}-matrix which combined skills 4 and 5. To test the 5×4 entry of Q_{mod} I created $Q_{test} = (Q_{mod}, e_5)$ as described above.

Using data generated from the Q-matrix, I estimated the Rasch problem difficulties and LLTM skill difficulties for the Q-, $Q_{mod}-$, and Q_{test}-matrices using ConQuest (Wu, Adams, and Wilson, 1998). ConQuest

[4]Note that the use of only 20 problems was a constraint of the estimation package that I am using for this part of my work and I am looking into ways to estimate more problems.

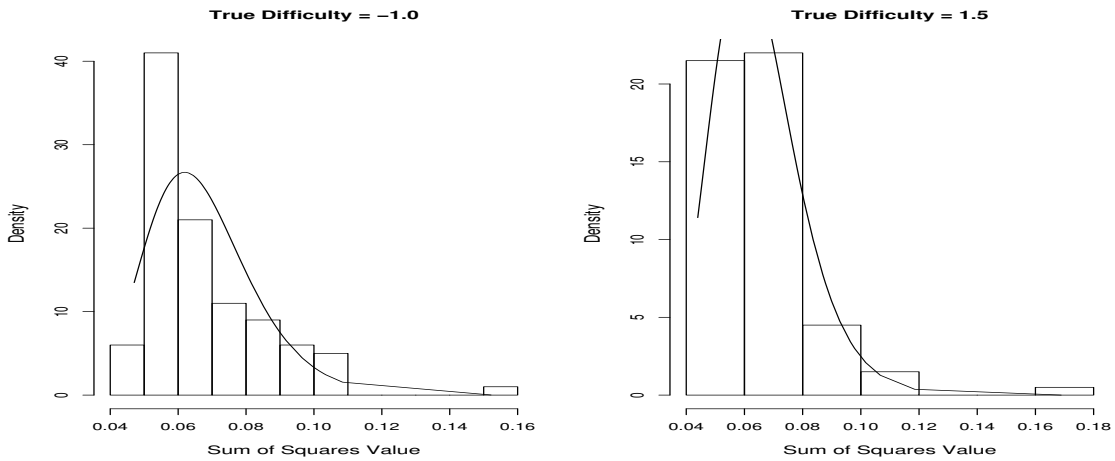


Figure 5: Sum of Squares Simulation Study 1

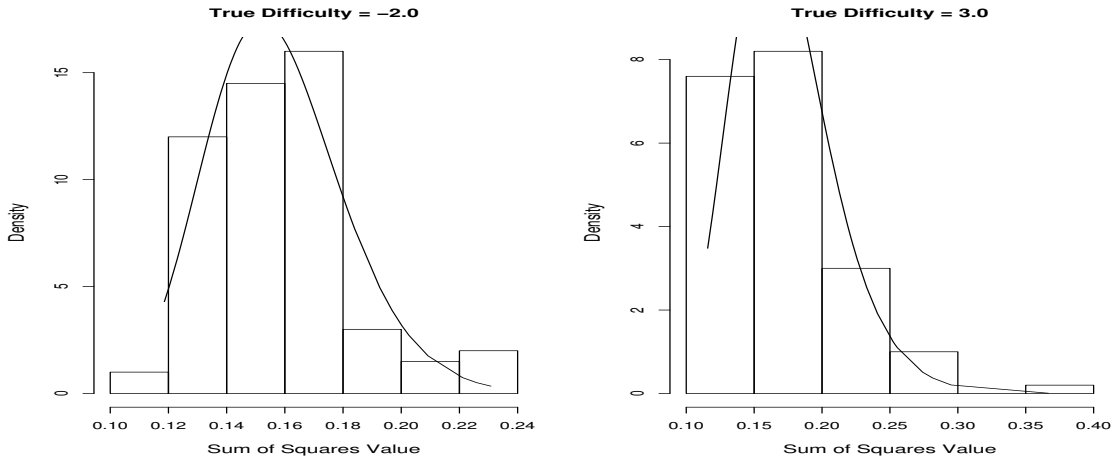


Figure 6: Sum of Squares Simulation Study 2

gives the deviance of the marginal model after integrating out θ and suggests comparing the difference in deviances between two models to compare the fit of the models. The difference has an asymptotic χ^2 distribution where the degrees of freedom is equal to the difference in the number of parameters estimated, in this case 1. Table 4 shows the deviance and number of parameters for the Q_{mod} and Q_{test} matrices. From this we can see that comparing deviances, a marginal likelihood ratio test, shows that Q_{test} gives a better fit. Using Equation 15, we would estimate

$$\hat{\sigma}_{5,4} = 1 + \frac{1.690}{1.180} = 2.43.$$

Figure 4 shows the Rasch vs. LLTM estimates for the original Q-matrix and Q_{mod}. In the plot for the Rasch vs. Q_{mod} estimates, one can note the spread of Q_{mod} estimates for the last skill. This skill is the combined skills 4 and 5 from the original Q-matrix. Noting this large spread, I decided to investigate the use of a sum of squares between the Rasch model and LLTM estimates in order to decide which skills should be

Table 5: Partial results from SS Study

True Difficulty	Estimated Location	Estimated Scale	KS-test p-value
−1.0	-2.727	0.235	0.098
1.5	-2.760	0.232	0.705
−2.0	-1.861	0.149	0.963
3.0	-1.781	0.214	0.144

further looked at for misfit.

$$SS = \sum_{\text{problems with same skill(s)}} (\text{Rasch estimate} - \text{LLTM estimate})^2$$

I have begun to look at the distribution of this SS statistic, under the null hypothesis that the LLTM is correct, in a pair of pilot simulation studies. In the first study, I considered the same "true" Q-matrix as in the simulation study above, 20 questions, 1000 examinees, skill difficulties ranging from -1 to 1.5, and $\theta \sim N(0, 1)$, replicated 100 times. I examined the histogram of the 100 SS for each of the five question groups implied by the Q-matrix. I considered a Gamma, Weibull, and Log-Normal distribution for the distribution of the SS. From the QQ-plots and Kolmogorov-Smirnov tests, it appeared as though a Log-Normal was giving the best fit. The second pilot study was designed and implemented in the same way, except that the skill difficulties were taken to be twice that of the first study in order to explore the affect of the size of the skill difficulty on the distribution of the SS. Figures 5 and 6 show the histogram of the 100 SS for the noted problem difficulty, overlaid with the estimated Log-Normal distribution. Table 5 shows the true difficulties, the Log-Normal estimated location and scale, and the Kolmogorov-Smirnov test p-values. The first two rows are from the first study and the last two rows are from the second study. In the Kolmogorov-Smirnov test the null hypothesis is that the data fit the given distribution. In each of the four cases, we see that we do not reject this claim. One can note that there are differences in the estimated locations between the two studies that does not appear to be due to the size of the true difficulty. I am currently designing a larger, more comprehensive simulation study to better understand the SS distribution when the LLTM is correct.

3.1.1 Other Testing Procedures

There are several reasonable alternatives to Fischer's method and the extension that I am working on. As an alternative, one could compare the fit of two models by comparing the BIC scores (Raftery, 1995) of the two models. It should be noted that this is the same score and test described above in Section 2.3. A test comparing BIC scores is similar to the marginal likelihood ratio test except that it adds a penalization to the models for increased complexity. In this situation we are testing the addition of a single skill to the Q-matrix and the number of free parameters to be estimated k will differ by one between the two models. Since the number of parameters varies by one, it is easy to see the fit-complexity trade-off inherent in the BIC score.

Table 4 also gives the BIC scores for the two models described in Section 3.1. Here the difference in BIC scores is about 30 and again we see that Q_{test} has a better fit. Thus all of the procedures discussed to compare models give the same answer.

3.2 Clustering

Another option for uncovering groups of problems is cluster analysis. Simply stated, clustering is the classification of problems into subsets where items within a subset share a common trait (in this case mathematical skills). With the set of Assistment problems being used there are several things that should be noted. Given

the broad range and uneven coverage of topics within the Assistment problems, we expect to have multiple clusters of varying sizes. With clustering I would like to be able to group problems that involve similar skills and be able to develop a meaningful set of skills. In this case meaningful would mean both leading to better predictions of exam scores and providing teachers with useful information about student knowledge.

An important step in cluster analysis is deciding what distance (i.e., Euclidean distance or Manhattan distance) to use to determine the similarity of two problems. Different distances will produce different clusters. In addition, one must decide what type of clustering to do. In hierarchical clustering, problems may belong to only one cluster. On the other hand, there exist clustering algorithms, such as ADCLUS (Shepard and Arabie, 1979), that allow overlapping of item clusters. In the Q-matrix used in the previous analysis roughly one-third of the problems are identified as having more than one skill. I think this is a indication that we need an algorithm in which problems may below to more than one cluster.

Clustering can use either a bottom-up (agglomerative) or top-down (divisive) approach. In a bottom-up approach single problem clusters are slowly merged into larger clusters. In a top-down cluster analysis all problems start in one large cluster and are sorted into smaller clusters. Since we already have a clustering based on the Q-matrix, I forsee starting with that and taking both a bottom-up and top-down approach. Figure 3 shows several cases where a top-down approach may be useful. In particular, the large cluster of problems coded with skill 36 appear to be in two smaller clusters. There are also examples where it may make sense to merge problems. Skills 45 (mean) and 46 (mode) each appear once at just below 2 on the Rasch problem difficulty scale and just under 0 on the LLTM problem difficulty scale. Since problem difficulty estimates are close under both models, the distinction between mean and mode may not be necessary.

Given the current Q-matrix and set of Assistment problems, I will propose an iterative algorithm to cluster the problems and skills. As a first step, I will apply biclustering to the given Q-matrix. Biclustering (Cheng and Church, 2002) allows for simultaneous grouping of problems and skills and also allows overlapping clusters. In addition, I will also cluster the response matrix so that items appear together if they have similar patterns of responses across students. These clusterings will be used in combination to do a stepwise hierarchical clustering which will alternate between both agglomerative and divisive steps. Since association between items caused by the student to student proficiency variation may overtake the clustering, I will condition out the student proficiency.

4 Accounting For Learning Over Time

In the models presented so far I have estimated a single student proficiency parameter. However, since students are working on the tutor over the course of several months we should be accounting for learning over time. During the year students are attending class and learning new topics, it is reasonable to assume that a student's proficiency will increase over time. This suggests that the proficiency parameters in the models should be adjusted to reflect this.

Much of the literature (Anderson, 1985; Embretson, 1991) concerning models with learning over time discusses multiple instances of the same questions or exams. However, in the case of the Assistment tutor data, we have few repeated items. In addition, students in the same class are not necessarily seeing the same problems during each session. One of the first choices we must make is deciding how to break down the time. For example, we could use each session, each week, or each month as a set time where we think student proficiency is relatively unchanging. Many students are only using the tutor once a week so the first two time periods are the same. I plan to start here, using one week's worth of work as a set of data. If the total number of problems seen is small, I may have to expand the time frame to obtain more data in order to

make conclusive statements. In addition, we must decide if we are going to look at individual problems or problems with the same set of skill(s).

Before we measure change, we must first consider the nature of change when students see problems (or skills) over multiple occasions. A simplex pattern, which accounts for performance in tasks where increasingly more (or alternatively fewer) abilities are needed, was noted by Corballis (1965) to account for intercorrelations between occasions. Consider the use of M abilities over K occasions. Assume that on the first occasion only one ability is used, but that each additional occasion depends on $k-1$ additional abilities. We can arrange this notion in a $K \times M$ (where K=M) matrix

$$A = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & & & \vdots \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix}.$$

Embretson (1991) uses this idea to develop a multidimensional Rasch model for learning and change (MRMLC). By imposing the simplex structure, we can account for individual differences at each occasion. This idea is then combine this with the Rasch model to obtain a multidimensional item response model

$$P(X_{i(k)j} = 1|\theta_{ik}, \beta_j, a_k) = \frac{e^{\sum_m a_{km}\theta_{im} - \beta_j}}{1 + e^{\sum_m a_{km}\theta_{im} - \beta_j}}, \tag{16}$$

where θ_{im} is the m^{th} ability of student i and β_j is the difficulty of problem j. Note that the problem difficulty remains the same over time and we can track student learning by looking at the change in θ over occasions.

Another option would be using a latent class analysis and track a student's learning by looking at how they progress through the different classes during the year. One possible measurement model is the mixture Rasch model (MRM; Rost, 1990) which assumes that a Rasch model holds within each latent class. In this model, each latent class has a different set of problem difficulties. In addition, members of each latent class may have different abilities. According to the MRM, the probability of a correct response is

$$P(X_{ij} = 1|g, \theta_{ig}) = \sum_{g=1}^{G} \pi_g \frac{1}{1 + e^{-(\theta_{ig} - \beta_{jg})}}, \tag{17}$$

where $g = 1, ..., G$ is an index for the latent class, $i = 1, ..., N$ are the students, $j = 1, ...J$ are the problems, π_g is the proportion of students in each class, θ_{ig} is the latent ability of student i within class g, and β_{jg} is the Rasch difficulty parameter of item j in class g.

When we combine the MRM with a latent transition analysis (LTA) model we can explicitly model student learning over time. Known as the LTA-MRM (Cho, Cohen, Kim, and Bottge, 2007) the probability of a correct response is now

$$P(X_{ijt} = 1|g_t, \theta^*_{igt}) = \sum_{g_1=1}^{G_1} \cdots \sum_{g_T=1}^{G_T} \pi_{g_1} \prod_{t=2}^{T} \tau^{(t-1)}_{g_t|g_{t-1}} \frac{1}{1 + e^{-(\theta^*_{itg_t} - \beta_{jtg_t})}}, \tag{18}$$

where g_t is an index for the latent class at time t, $\theta^*_{itg_t}$ is the ability of student i within pattern g_t, β_{jtg_t} is the difficulty of item j at time t for pattern g_t, π_{g_1} is the proportion of students in latent class g_1 at time 1, and $\tau^{(t-1)}_{g_t|g_{t-1}}$ is the transition probability from latent class g_{t-1} at time (t-1) to latent class g_t at time t. Cho, Cohen, Kim, and Bottge (2007) suggest using a stationary Markov chain to model the transition probabilities

between latent classes. Let the class at time t be denoted at y_t for $t = 2, ...T$ with G possible classes. The transition probability is then

$$p(y_t = g_t | z_{t-1} = g_{t-1}) = p_{g_t g_{t-1}}(t) = p_{g_t g_{t-1}}(t+1) = p_{g_t g_{t-1}}, \tag{19}$$

where $\sum_{g_{t-1}} p_{g_t g_{t-1}} = 1$.

5 Compare Modeling Approaches

A third goal, depending on time and the outcome of the previous two, is comparing different modeling approaches. When modeling exam or tutor response data, there are many assumptions that go into choosing a model. We need to make decisions about how to model both the latent variables and the skills. Do we want a single continuous variable for student proficiency or should divide student proficiency into separate discrete variables according to the skills? Should we model skills additively (as in the LLTM in Section 2.2) or should we use a conjunctive model? One must keep in mind that while a finer grained skills model may help us when giving feedback to teachers, it also means that we will lose reliability when making inference since there will be fewer observations for each skill.

Below I will describe a few of the many models that may be considered when modeling response data. In each of the following models

$$X_{ij} = 1/0 \qquad \text{indicating whether student } i \text{ answer question } j \text{ correctly}$$

and

$$q_{jk} = \begin{cases} 1 & \text{if problem } j \text{ contains skill } k \\ 0 & \text{else} \end{cases}$$

The first model is the DINA (deterministic inputs, noisy "and" gate; Junker and Sijtsma, 2001) model. In the DINA model we make a conjunctive assumption about the skills, which is different from the additive skill assumption in the LLTM. In a conjunctive model, a student must possess all the skills required by a problem in order to answer correctly. Thus, in the case of the DINA model we look at an individual student's skill level knowledge instead of the overall skill difficulty as in the LLTM. First denote

$$\alpha_{ik} = 1/0 \qquad \text{indicating whether or not student } i \text{ possesses skill } k.$$

In the DINA model, we define the latent response variables as

$$\xi_{ij} = \prod_{k:q_{jk}=1} \alpha_{ik} = \prod_{k=1}^{K} \alpha_{ik}^{q_{jk}}$$

We then relate the latent response variables ξ_{ij} to the observed responses X_{ij} through the slip s_j (having all the skills but answering incorrectly) and guess g_j (not having all the skills but answering correctly) parameters:

$$P(X_{ij} = 1) = \begin{cases} 1 - s_j & \text{if } \xi_{ij} = 1 \\ g_j & \text{if } \xi_{ij} = 0 \end{cases}$$

Thus, the joint likelihood for all responses under the DINA model is

$$P(\underline{X} = \underline{x}) = \prod_{i=1}^{N} \prod_{j:i \text{ saw } j} \left[s_j^{1-X_{ij}} (1 - s_j)^{X_{ij}} \right]^{\xi_{ij}} \left[(1 - g_j)^{1-X_{ij}} g_j^{X_{ij}} \right]^{1-\xi_{ij}}$$

Estimation for the DINA model can be done with MCMC methods (Anozie and Junker, 2007).

The next model, a multi-dimensional item response theory (MIRT) model, is similar to the Rasch model except that we now have a vector of student proficiency parameters that measure different abilities. In the case of the MCAS exam we could use the five strands from which problems are drawn: Number Sense and Operations; Patterns, Relations, and Algebra; Geometry; Measurement; and Data Analysis, Statistics, and Probability. The MIRT model may simply be represented as

$$P_j(\underline{\theta}_i) = P(X_{ij} = 1 | \theta_1, ..., \theta_d, \beta_j) = P(a_{j1}\theta_1 + a_{j2}\theta_2 + \cdots + a_{jd}\theta_d - \beta_j) \tag{20}$$

where the probability can be a logistic function or any of several response functions. This can be compared to Embretson's (1991) multi-dimensional Rasch model for learning and change (MRMLC), discussed in Section 4, which allows for different abilities at different times.

When comparing different modeling approaches there are several areas in which I will focus. For the Assistment data I am interested in whether a single student ability parameter or a multi-dimensional one does better. While a more general single student ability estimate does not give as much detail, there may not be enough information or observations to estimate several different abilities. I would like to know if the extra effort of estimating multiple abilities leads to better prediction models. In addition, I am interested in finding a model which accurate estimates student skill knowledge. Hopefully this will be a model that both helps improve predictions but that also yields beneficial information for teachers.

6 Summary of Proposed Work

This proposal discusses three topic areas in which I would like to make improvements to the current educational research methods. First, I want finish my work on methods to improve the Q-matrix. Second I plan to look at models which account for learning over time. Third I plan to compare various modeling approaches to add to the current discussion of the applicability and the pros and cons of each.

The first topic, making improvements to the Q-matrix is the one I will work on first and has the highest priority in my work. Currently this work involves developing a method and test statistic to decide which skills to further investigate. I will also look at this task from a clustering perspective. In addition, this area has a practical application in naming and using the skills. I think that a meaningful and predictive set of skills is needed to move forward and make comments on the remaining two topics. The correct set of skills is needed both for feedback to teachers and for modeling student performance.

Although I have decided to work on it second, accounting for learning over time is an almost equally important task. If I decide to use a latent class analysis, it is likely there will be a separate Q-matrix for each latent class. In this case the first two areas will be merged. The final tasks of comparing modeling approaches is last on my list and has a lower priority. It is something I would like to do, but will scale it depending on time and advances in the first two areas.

References

Anderson, E.B. (1985). Estimating latent correlations between repeated testings. *Psychometrika*, 50, 3-16.

Anozie, N. O.; and Junker, B. W. (2007). *Investigating the utility of a conjunctive model in Q-matrix assessment using monthly student records in an online tutoring system.* National Council on Measurement in Education (NCME-07), April 12, 2007, Chicago, IL.

Anozie, N. O.; and Junker, B.W. (2006). Predicting end-of-year accountability assessment scores from monthly student records in an online tutoring system. *Proceedings of the American Association for Artificial Intelligence Workshop on Educational Data Mining (AAAI-06), July 17, 2006, Boston, MA*. Menlo Park, CA: AAAI Press. pp. 1-6. Technical Report WS-06-05.

Ayers, E. and Junker, B.W. (2007). IRT Modeling of Tutor Performance to Predict End-of-year Exam Scores. Under Revision for *Educational and Psychological Measurement.*

Barnes, T.M. (2003). *The Q-matrix Method of Fault-tolerant Teaching in Knowledge Assessment and Data Mining.* Ph.D. Dissertation, Department of Computer Science, North Carolina State University.

Bechger, T.M., Verstralen, H.H.F.M., and Verhelst, Norman D. (2002). Equivalent Linear Logistic Test Models. *Psychometrika*, 67, 123-136.

Bechger, T.M., Verstralen, H.H.F.M., Verhelst, N.D., and Maris, Gunter. (2004). Equivalent LLTMS: A rejoinder. *Psychometrika*, 69, 317-318.

Bishop, J.H. (1998). The Effect of Curriculum-Based External Exit Exam Systems on Student Achievement. *The Journal of Economic Education*, Vol 29, Issue 2, 171-182.

Cheng, Y and Church, G.M. (2000). Biclustering of Expression Data. In *Proceedings of the Eighth International Conference on Intelligence Systems for Molecular Biology*, 57-66. Menlo Park, Calif.: AAAI Press.

Cho, Sun-Joo, Cohen, Allan S., Kim, Seock-Ho, and Bottge, Brian. (2007). *Latent Transition Analysis With a Mixture item Response Theory Measurement Model.* Presented at the 2007 Annual Meeting of the National Council on Research in Education, Chicago, IL, April 2007.

Corballis, M.C. (1965). Practice and the simplex. *Psychological Review*, 72, 399-406.

Cronbach, L.J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika*, 16, 297-334.

De Boeck, P. and Wilson, M. (2004). *Statistics for Social Science and Public Policy*. New York: Springer.

Embretson, S.E. (1984). A General Latent Trait Model for Response Processes. *Psychometrika*, 49, 175-186.

Embretson, S.E. (1991). A Multidimensional Latent Trait Model for Measuring Learning and Change. *Psychometrika*, 56, 495-515.

Feng, M., Heffernan, N.T., and Koedinger, K.R. (2006). Predicting state test scores better with intelligent tutoring systems: developing metrics to measure assistance required. In Ikeda, Ashley and Chan (Eds.). *Proceedings of the 8th International Conference on Intelligent Tutoring Systems.* Springer-Verlag: Berlin. pp. 31-40.

Fischer, G.H. (1974). *Einführung in die Theorie psychologischer Tests: Grundlagen und Anwendungen.* (Introduction to the Theory of Psychological Tests: Foundations and Applications) Switzerland: Verlag Hans Huber.

Fischer, G.H. and Molenaar, I.W. (1995). *Rasch Models: Foundations, Recent Developments, and Applications*. New York: Springer Verlag.

Fischer, Gerhard H. (2004). Remarks On "Equivalent Linear Logistic Test Models" by Bechger, Verstralen, and Verhelst (2002). *Psychometrika*, 69, 305,315.

Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models. *Bayesian Analysis*,

1, Number 3, pp. 515-533.

Gelman, A., Carlin, J., Stern, H., and Rubin, Donald B. (2004). *Bayesian Data Analysis*, 2nd edn. New York: Chapman and Hall.

Gelman, A., Meng, X.-L., and Stern, H. (1996). Posterior Predictive Assessment of model fitness via realized discrepancies [with discussion]. *Statistica Sinica*, 6, 733-807.

Haist, S.A., Witzke, D.B., Quinlivan, S., Murphy-Spencer, A., and Wilson, J.F. (2003). Clinical Skills as Demonstrated by a Comprehensive Clinical Performance Examination: Who Performs Better - Men or Women? *Advances in Health Sciences Education*, 8: 189-199.

Heffernan, N.T., Koedinger, K.R. and Junker, B.W. (2001). *Using Web-Based Cognitive Assessment Systems for Predicting Student Performance on State Exams.* Research proposal to the Institute of Educational Statistics, US Department of Education. Department of Computer Science at Worcester Polytechnic Institute, Worcester County, Massachusetts.
http://nth.wpi.edu/pubs_and_grants/Grant_to_IES_with_WPS.pdf

Janssen, R. and De Boeck, P. (2006). *A random-effects version of the LLTM*. Technical report, Department of Psychology, University of Leuven, Belgium.

Junker, B. W. (2006). Using on-line tutoring records to predict end-of-year exam scores: experience with the ASSISTments Project and MCAS 8th grade mathematics. To appear in Lissitz, R. W. (Ed.), *Assessing and modeling cognitive development in school: intellectual growth and standard setting.* Maple Grove, MN: JAM Press.

Junker, B.W. and Sijtsma K. (2001). Cognitive Assessment Models with Few Assumptions and Connections with Nonparametric Item Response Theory. *Applied Psychological Measurement*, 25: 258-272.

Lord, F.M. and Novick, M.R. (1968). *Statistical Theories of Mental Test Scores*. Reading, MA: Addison-Wesley.

Maccini, P. and Hughes, C.A. (2000). Effects of a problem-solving strategy on the introductory algebra performance of secondary students with learning disabilities. *Learning Disabilities Research and Practice*, 15, 10-21.

Massachusetts Department of Education. (2004). *2004 MCAS Technical Report*. Downloaded December 2005 from http://www.doe.mass.edu/mcas/2005/news/04techrpt.pdf.

Massachusetts Department of Education. (2006). *School and District Accountability.* Retrieved April 2006 from http://www.doe.mass.edu/sda/.

Nuthall, G. and Alton-lee, A. (1995). Assessing Classroom Learning: How Students Use Their Knowledge and Experience to Answer Classroom Achievement test Questions in Science and Social Studies. *American Educational Research Journal*, Vol. 31, No. 1, 185-223.

Olson, L. (2005). State Test Programs Mushroom as NCLB Mandate Kicks In. *Education Week*, Nov. 30: 10-14.

Pardos, Z.A., Heffernan, N.T., Anderson, B. and Heffernan, C.L. (2006). *Using Fine-Grained Skill Models to Fit Student Performance with Bayesian Networks.* Workshop in Educational Data Mining held at the Eighth International Conference on Intelligent Tutoring Systems. Taiwan. 2006. http://web.cs.wpi.edu/Research/trg/public/project/papers/its06/zpardos-its-final22.pdf

Protter, M.H. and Morrey, C.B. Jr. (1991). *A First Course in Real Analysis. 2nd Edition.* New York: Springer.

R Development Core Team (2004). *R: A language and environment for statistical computing.* R Founda-

tion for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.

Raftery, A.E. (1995). Bayesian Model Selection in Social Research. *Sociological Methodology*, Vol. 25, 111-163.

Rost, J. (1990). Rasch models in latent classes: An integration of two approaches to item analysis. *Applied Psychological Measurement*, 14, 271-282.

Schofield, L., Taylor, L. and Junker, B.W. (2006). *The use of cognitive test scores in evaluating black-white wage disparity.* Working paper.

Shepard, R.N., and Arabie, P. (1979). Additive clustering: Representation of Similarities as Combinations of Discrete Overlapping Properties. *Psychological Review*, 2, 87-123.

Spiegelhalter, D.J., Thomas, A. and Best, N.G. (2003). *WinBUGS: Bayesian Inference Using Gibbs Sampling, Manual Version 1.4*. Cambridge: Medical Research Council Biostatistics Unit.

Tatsuoka, Kikumi K. (1995). Architecture of Knowledge Structures and Cognitive Diagnosis: A Statistical Pattern Recognition and Classification Approach. Chapter 14 in P.D. Nichols, S.F. Chipman, and R.L. Brennan. (Eds). *Cognitively Diagnostic Assessment* (1995). Hillsdale, NJ: Lawrence Erlbaum Associates.

Tatsuoka, Kikumi K. (1983). Rule Space: An Approach for Dealing with Misconceptions Based on Item Response Theory. *Journal of Educational Measurement*. Vol. 20, No. 4, 345-354.

van de Vijver, Fons J.R. (1988). Systematizing the Item Content in Test Design. Chapter 13 in R. Langeheine and J. Rost. (Eds). *Latent Trait and Latent Class Models* (1988). New York: Plenum Press.

van der Linden, W.J. and Hambleton, R.K. (1997). *Handbook of Modern Item Response Theory*. New York: Springer-Verlag.

Wright, B.D. (1995). 3PL or Rasch? *Rasch Measurement Transactions*, 9(1), 408-409.

Wu, Margaret L., Adams, Raymond J., and Wilson, Mark R. (1998). *ACER ConQuest: Generalised item response modeling software*. Melbourne, Australia: Australian Council for Educational Research.